

**No English title available.**

Patent Number: DE19614789  
Publication date: 1997-09-25  
Inventor(s): JUENGST ERNST-WERNER DR PHIL (DE); HEINRICH MICHAEL DIPLO PHYS (DE)  
Applicant(s): DAIMLER BENZ AG (DE)  
Requested Patent:  [DE19614789](#)  
Application Number: DE19961014789 19960406  
Priority Number(s): DE19961014789 19960406  
IPC Classification: G06F15/18; G06F17/50  
EC Classification: [G06F17/60C4](#)  
Equivalents:  [EP0896702](#) (WO9738384),  B1, JP11507153T, JP3059763B2,  [WO9738384](#)

---

**Abstract**

---

The aim of the invention is to provide an automatic, resource-oriented method of configuring modular systems which configures a system by comparing resources of the same type but of different quality for compatibility, which is easy to monitor, as well as being efficient, flexible and suitable for use in a wide range of applications and which enables information of the types of component listed in a catalogue to be easily updated when changes are made or the component range expanded. In the method proposed, the following information is explicitly defined and stored: the different qualities which the resources defined in the catalogue listing the types of component have, how the resources having different qualities are compared to each other in order to decide which of the resources provided are compatible with resources of a particular quality requested. A system is configured by repeatedly comparing the requested resources with those provided with regard to type, size and compatibility. The method proposed stipulates how components are checked automatically for compatibility and how other components are selected if the required resources are not available.

---

Data supplied from the **esp@cenet** database - I2

**THIS PAGE BLANK (USPTO)**

OOP 22762  
⑯ BUNDESREPUBLIK

DEUTSCHLAND



DEUTSCHES

PATENTAMT

⑯ Patentschrift

⑯ DE 196 14 789 C 1

⑯ Int. Cl. 6:

G 06 F 15/18

G 06 F 17/50

31  
DE 196 14 789 C 1

- ⑯ Aktenzeichen: 196 14 789.1-53  
⑯ Anmeldetag: 6. 4. 96  
⑯ Offenlegungstag: —  
⑯ Veröffentlichungstag der Patenterteilung: 25. 9. 97

Innerhalb von 3 Monaten nach Veröffentlichung der Erteilung kann Einspruch erhoben werden

⑯ Patentinhaber:

Daimler-Benz Aktiengesellschaft, 70567 Stuttgart,  
DE

⑯ Erfinder:

Jüngst, Ernst-Werner, Dr. phil. nat., 13349 Berlin, DE;  
Heinrich, Michael, Dipl.-Phys., 14052 Berlin, DE

⑯ Für die Beurteilung der Patentfähigkeit in Betracht gezogene Druckschriften:

DE	39 11 465 C2
WO	96 28 784 A1
WO	94 23 372 A1

⑯ Verfahren zur automatischen Konfigurierung eines technischen Systems unter Berücksichtigung unterschiedlicher Qualitäten von Komponenten-Außeneinwirkungen

⑯ Aufgabe der Erfindung ist, ein Verfahren zur automatischen Ressourcen-orientierten Konfigurierung modularer technischer Systeme zu schaffen,  
- das beim Konfigurieren Ressourcen gleicher Art, aber mit unterschiedlichen Qualitäten auf Kompatibilität vergleicht,  
- das überschaubar und doch wirksam sowie flexibel und universell einsetzbar ist  
- und das es ermöglicht, die Informationen im Katalog verfügbarer Komponententypen bei Änderungen und Erweiterungen im Komponentenspektrum mit geringem Aufwand nachzuführen.

Beim erfindungsgemäßen Verfahren werden folgende Informationen explizit definiert und gespeichert:

- Welche unterschiedlichen Qualitäten haben die im Katalog verfügbarer Komponententypen definierten Ressourcen?
- Wie werden Ressourcen mit unterschiedlichen Qualitäten miteinander verglichen, um zu entscheiden, welche bereitgestellten Ressourcen mit nachgefragten Ressourcen einer bestimmten Qualität kompatibel sind?

Bei der Konfiguration werden wiederholt die geforderten Ressourcen hinsichtlich Art, Umfang und Kompatibilität mit den bereitgestellten Ressourcen verglichen. Das erfindungsgemäße Verfahren legt fest, wie automatisch auf Kompatibilität geprüft wird und wie bei nicht erfüllter Ressourcenbilanz weitere Komponenten ausgesucht werden.

DE 196 14 789 C 1

## Beschreibung

Die Erfindung betrifft ein Verfahren gemäß des Oberbegriffs des Anspruchs 1.

Der Stand der Technik kennt mehrere Verfahren und Einrichtungen, die automatisch ein technisches System aus Komponenten konfigurieren. Bei diesen Verfahren sind folgende Informationen vorgegeben: die formale Spezifikation des zu konfigurierenden Systems sowie ein Katalog von Komponenten-Typen und ihren Wechselwirkungen in rechnerverfügbarer Form. Als gewünschtes Ergebnis liefern die Verfahren mindestens eine Stückliste, d. h. eine Auflistung aller verwendeten Komponententypen und die Anzahl der Exemplare jedes verwendeten Typs — oder die Aussage, daß die Spezifikation mit den verfügbaren Komponententypen nicht erfüllbar ist.

Den Katalog stellt ein menschlicher Entwickler einmal auf und aktualisiert ihn bei der Einführung neuer oder der Streichung alter Komponententypen. Jedesmal, wenn ein technisches System zu entwerfen ist, formalisiert ein menschlicher Projektierer die Anforderungen an das System in einer Form, die die Informationsverarbeitungseinrichtung automatisch verarbeiten kann. Dann erzeugt die Einrichtung automatisch die Konfiguration des technischen Systems.

Ältere aus der Literatur bekannte Ansätze, diese Aufgabe automatisch zu lösen, basieren darauf, daß ein Entwickler direkt Beziehungen zwischen Komponententypen notiert und daß diese abgespeicherten Relationen beim Konfigurieren ausgewertet werden. Zwei Beispiele: Ein Verbraucher vom Typ x benötigt eine Spannungsquelle vom Typ y. Eine Komponente vom Typ A ist unvereinbar mit einer solchen vom Typ B. Damit die technischen Abhängigkeiten automatisch berücksichtigt werden, ist ein umfangreiches Netz von IF–THEN-Regeln aufzustellen. Dieses Netz ist schwer zu überblicken. Die Gründe für ein bestimmtes Konfigurerungsergebnis sind kaum nachzuvollziehen. Außerdem veraltet es bei der Einführung neuer Komponententypen leicht. Diese Ansätze werden beispielsweise in

J. McDermott: "R1: A Rule-Based Configurer of Computer Systems", Artificial Intelligence 19 (1982), S. 39–88 und

E. Lehmann: "SICONFEX, ein Expertensystem für die Konfiguration eines Betriebssystems", 15. Jahrestagung der GI, 1985, S. 792–805

vorgestellt.

Eine Klasse von Lösungen, die diese Nachteile vermeidet, ist aus DE 39 11 465 A1/C2 sowie

A. Schwancke und J. P. Bernert: "Ressourcenorientierte Konfigurierung von Kommunikationssystemen", in N. Kratz, A. Günter & J. Hertzberg (Hrsg.): "Beiträge zum 4. Workshop Planen und Konfigurieren", Forschungsinstitut für anwendungsorientierte Wissensverarbeitung, Mai 1990, S. 157–170, und M. Heinrich: "Ressourcen-orientierte Modellierung als Basis des Konfigurerens modularer Technischer Systeme", in A. Günter & R. Cunis (Hrsg.) "Beiträge zum 5. Workshop Planen und Konfigurieren", Labor für Künstliche Intelligenz der Universität Hamburg, April 1991, Seiten 61 bis 74 und

B. Stein & J. Weiner: "MOKON – Eine modellbasierte Entwicklungsplattform zur Konfigurierung technischer Anlagen", in A. Günter & R. Cunis

(Hrsg.): "Beiträge zum 5. Workshop Planen und Konfigurieren", Labor für Künstliche Intelligenz der Universität Hamburg, April 1991, Seiten 100 bis 106 bekannt: das ressourcenorientierte Konfigurieren. Beim ressourcenorientierten Konfigurieren werden im Katalog der verfügbaren Komponententypen nicht direkt Beziehungen zwischen Komponenten beschrieben. Vielmehr ist im Katalog der Komponententypen für jeden Typ angegeben, welche Ressourcen (Art und Umfang) eine Komponente des Typs bereitstellt und welche Ressourcen sie fordert, das heißt verbraucht, benutzt, belegt oder voraussetzt. Mit Hilfe von Ressourcen werden also die Außenwirkungen von Komponenten formalisiert dargestellt. Ressourcen beschreiben typischerweise physikalische Größen, Eigenschaften oder Schnittstellen. Bsp.: Strom bei 12-V-Gleichspannung oder Steckplatz in einem Magazin. Die Anforderungen an das zu konfigurerende technische System werden durch Angabe derjenigen Ressourcen formal beschrieben, die das technische System bereitzustellen hat.

Manche Ansätze kombinieren das ressourcenorientierte Konfigurieren mit Verfahren, die auf expliziten Nebenbedingungen ("constraints") beruhen. In WO 94/23372 A1 und WO 96/28784 A1 wird das zu konfigurerende technische System formal spezifiziert durch eine Anforderungsliste ("list of requests"). Eine Anforderung kann eine Ressource (im Sinne des ressourcen-orientierten Konfigurerens), eine explizit genannte Komponente oder eine Nebenbedingung sein. Eine Nebenbedingung beschreibt eine bestimmte Beziehung zwischen mehreren Komponenten, die das zu konfigurerende technische System erfüllen muß. In WO 94/23372 A1 und WO 96/28784 A1 werden drei Arten von Nebenbedingungen beschrieben:

- Eine Komponente muß physikalisch in einer anderen Komponente enthalten sein ("container constraint")
- Eine Komponente muß mit einer anderen Komponente verbunden sein ("connection constraint")
- Die Verwendung einer Komponente erfordert die Verwendung einer anderen Komponente ("component constraint").

Außerdem wird in WO 94/23372 A1 und WO 96/28784 A1 die Verwendung einer weiteren Relation beschrieben, die dort Funktionshierarchie ("functional hierarchy") genannt wird. Diese Art von Relation wird in der Literatur üblicherweise als Taxonomie oder Verwandtschaftshierarchie unter Komponententypen bezeichnet, siehe beispielsweise DE 39 11 465 C2.

Daß zur Konfigurierung Nebenbedingungen und eine Taxonomie ausgewertet wird, ist auch aus anderen Literaturstellen bekannt, z. B. aus R. Cunis, A. Günter, I. Syska, H. Bode & H. Peters: "PLAKON – Modellierung von technischen Domänen mit BHIBS". In: M. Hein, W. Tank & J. Hertzberg (Hrsg.): "Beiträge zum 3. Workshop Planen und Konfigurieren", Arbeitspapiere der Gesellschaft für Mathematik und Datenverarbeitung 388, Mai 1989, Seiten 33 bis 48. Dort wird auch die Verwendung von Aufbau-Hierarchien ("part-of hierarchies") beschrieben, die den in WO 94/23372 A1 und Wo 96/28784 A1 erwähnten "container constraints" entsprechen.

Nach dem Stand der Technik werden Anforderungen und Bereitstellungen einer Ressource nach Art und Um-

fang verglichen. Falls beispielsweise beim Konfigurieren eine Komponente V ausgewählt wird, die Strom mit 12-V-Gleichspannung benötigt, so wird danach eine andere Komponente B ausgewählt werden, die Strom mit 12-V-Gleichspannung bereitstellt und damit die Ressourcenbilanz hinsichtlich dieser Ressource ausgleicht.

Diese Auswahl ist aber nicht in jedem Falle geeignet, die Anforderung zu befriedigen: Falls die Komponente V Strom bei 12-V-Gleichspannung in einer bestimmten Spannungsstabilität benötigt, beispielsweise zwischen 10 V und 14 V Ist-Spannung, so muß B diese Ressource mit mindestens der geforderten Spannungsstabilität bereitstellen. Der Stand der Technik liefert zwar ein effizientes Verfahren, um Ressourcen hinsichtlich Art und Umfang zu vergleichen, jedoch kein geeignetes Verfahren, mit dem ein System zur automatischen Konfigurierung unterschiedliche Qualitäten von Ressourcen unterscheiden kann und entscheiden kann, ob die Anforderungen nach einer Ressource in der erforderlichen Qualität abgedeckt werden.

Eine Abhilfe nach dem Stand der Technik ist die, für jede mögliche Qualität einer Ressource formal eine eigene Ressource in den Katalog verfügbarer Komponententypen aufzunehmen. Im obigen Beispiel muß mit der Ressource "Strom bei 12-V-Gleichspannung mit Ist-Spannung zwischen 10 V und 14 V" anstelle mit "Strom bei 12-V-Gleichspannung" gearbeitet werden. Dadurch wird die Zahl der erforderlichen Ressourcen um Größenordnungen höher. Ein weiterer und prinzipieller Nachteil: Falls im Katalog verfügbarer Komponententypen kein Spannungsversorger enthalten ist, der Strom bei 12-V-Spannung genau dieser Stabilität bereitstellt, so kann die Konfigurierungsaufgabe nicht gelöst werden – beispielsweise auch dann nicht, wenn ein anderer Komponententyp existiert, der Strom bei 12-V-Spannung mit höherer Stabilität bereitstellt und den daher ein menschlicher Projektierer auswählen würde, da er diese Ressourcenanforderung abdecken würde.

Will man diesen Nachteil vermeiden, bleiben nach dem Stand der Technik nur zeitaufwendige und fehlerträchtige Wege, um trotzdem die Konfigurierungsaufgabe in der geforderten Genauigkeit zu lösen, beispielsweise:

- Man kombiniert ein Verfahren des ressourcenorientierten Konfigurierens mit herkömmlichen Modellierungsmethoden, die explizit Beziehungen zwischen Komponententypen enthalten. Ein Beispiel: Man stellt IF–THEN-Regeln der Form "Komponententyp A ist unvereinbar mit Komponententyp B" auf. Dieser Ausweg erfordert einen deutlich gesteigerten Pflegeaufwand bei Einführung neuer Typen von Komponenten.

- Man formuliert Regeln, die die Kompatibilität zwischen unterschiedlichen Qualitäten einer Ressource beschreiben. Dies führt ebenfalls zu einer großen Anzahl zusätzlicher Regeln.

- Man ändert bei jeder neuen Konfigurierungsaufgabe die Spezifikation des zu konfigurierenden technischen Systems so ab, daß die passenden Komponenten ausgewählt werden. Bei Aufnahme neuer Komponententypen in den Katalog verknüpft man den Komponententyp so mit Ressourcen, daß die Komponenten wie gewünscht zueinander passen. In der Regel wird die Aufnahme eines neuen Komponententyps dann erfordern, daß große Teile des bestehenden Katalogs überarbeitet werden.

Bei diesem Vorgehen kann also die formale Spezifikation nicht nur aufgrund der technischen Anforderungen an das zu konfigurierende technische System aufgestellt werden, sondern hängt zusätzlich von den verfügbaren Komponententypen ab. Auch bei diesem Vorgehen ist der Pflegeaufwand erheblich, und de facto wird keine Einrichtung zum automatisches Konfigurieren mehr geschaffen.

- Man formuliert für jede solche Situation spezielle IF–THEN-Regeln, manchmal auch als Produktionsregeln bezeichnet. Diese Regeln sind notgedrungen anwendungsspezifisch. Beispielsweise stellt man Regeln für die Ressource Strom bei 12-V-Gleichspannung und für unterschiedliche Spannungsstabilitäten auf.

Ein anderer Ausweg nach dem Stand der Technik ist der, durch die automatische Konfigurierung nur einen groben Entwurf durchführen zu lassen. Um den Grobentwurf nachträglich zu einer genauen Konfiguration des technischen Systems zu verfeinern, gibt es zwei Alternativen:

- Man wählt von Hand die tatsächlich verwendeten Komponenten aus
- Konventionelle Ansätze des Konfigurierens werden ergänzend eingesetzt, beispielsweise ein System, das mit IF–THEN-Regeln die Auswahl der Komponenten explizit steuert und daher sehr anwendungsspezifisch ist.

Alle diese Abhilfen führen dazu, daß man die Vorteile des ressourcenorientierten Konfigurierens weitgehend wieder einbüßt.

Aufgabe der Erfindung ist es, ein Verfahren zur automatischen ressourcenorientierten Konfigurierung modularer technischer Systeme zu schaffen,

- das beim Konfigurieren Ressourcen gleicher Art, aber mit unterschiedlichen Qualitäten daraufhin vergleicht, ob sie miteinander kompatibel sind,
- das überschaubar und doch wirksam sowie flexibel und universell einsetzbar ist
- und das es ermöglicht, die Informationen im Katalog verfügbarer Komponententypen bei Änderungen und Erweiterungen im Komponentenspektrum mit geringem Aufwand nachzuführen.

Diese Aufgabe wird bei einem Verfahren der eingangs genannten Art mit den im Anspruch 1 gekennzeichneten Merkmalen gelöst.

Das erfindungsgemäße Verfahren zeichnet sich dadurch aus, daß folgende Informationen explizit definiert und gespeichert werden:

- Welche unterschiedlichen Qualitäten haben die im Katalog verfügbarer Komponententypen definierten Ressourcen?
- Wie werden Ressourcen mit unterschiedlichen Qualitäten miteinander verglichen, um zu entscheiden, welche bereitgestellten Ressourcen mit nachgefragten Ressourcen einer bestimmten Qualität kompatibel sind?

Diese Informationen werden nicht den Ressourcen selber, sondern den Ressourcenarten, nach denen die Ressourcen klassifiziert werden, zugeordnet. Die Informationen für eine Ressourcenart gelten für alle Res-

sourcen dieser Art. Die Informationen über Ressourcenarten werden getrennt von denen über Komponententypen gespeichert und gepflegt.

Im Verlaufe einer ressourcenorientierten Konfigurierung wird wiederholt eine Ressourcenbilanz aufgestellt, und in ihr werden die geforderten Ressourcen nicht nur hinsichtlich Art und Umfang mit den bereitgestellten Ressourcen verglichen, sondern auch auf Kompatibilität mit jenen untersucht. Die Bilanz ist nur dann ausgeglichen, wenn jede Anforderung nach einer Ressource entweder durch die Bereitstellung einer Ressource mit der gleichen Qualität oder einer kompatiblen Ressource abgedeckt wird. Um zu entscheiden, ob eine bereitgestellte Ressource einer bestimmten Qualität kompatibel ist mit einer geforderten Ressource der gleichen Art, aber einer anderen Qualität, wendet die Informationsverarbeitungsanlage automatisch eine Entscheidungsvorschrift an, die im Speicher für Ressourcenarten abgelegt ist. Kann eine geforderte Ressource weder durch eine bereitgestellte Ressource gleicher Qualität noch durch eine bereitgestellte kompatible Ressource abgedeckt werden, so sind weitere Komponenten auszuwählen, die die erforderlichen Ressourcen in der benötigten Qualität oder kompatible Ressourcen bereitstellen.

#### Beschreibung der vorteilhaften Ausgestaltung

Ressourcen sind nach Ressourcenarten klassifiziert, z. B. die Ressourcenarten "Strom bei 12-V-Gleichspannung" oder "Steckplatz".

Ein Rechner zur automatischen Konfigurierung eines technischen Systems, der das erfundungsgemäße Verfahren anwendet, hat Lesezugriff auf folgende Speicher:

— Der erste Speicher enthält den Katalog verfügbarer Komponententypen, der gemäß des Oberbegriffs der Erfindung folgende Informationen notiert:

- Art, Umfang und Qualitäten der Ressourcen, die eine Komponente des Typs bereitstellt
- Art, Umfang und Qualitäten der Ressourcen, die eine Komponente des Typs fordert, d. h. verbraucht oder belegt oder benutzt oder voraussetzt.

— Der zweite Speicher enthält die Beschreibung der Ressourcenarten. Sind für eine Ressourcenart unterschiedlichen Qualitäten zu berücksichtigen, so sind in diesem Speicher folgende Informationen notiert:

- Welche unterschiedlichen Qualitäten haben die bereitgestellten und die geforderten Ressourcen dieser Art?
- Mit welcher Vorschrift wird eine bereits bereitgestellte oder eine bereitstellbare Ressource R1 dieser Art daraufhin geprüft, ob sie mit einer geforderten Ressource R2 derselben Art kompatibel ist?

— Im dritten Speicher ist die formale Spezifikation des zu konfigurierenden technischen Systems abgelegt, die Art, Umfang und Qualitäten derjenigen Ressourcen festlegt, die das technische System bereitstellen soll.

Gemäß dem erfundungsgemäßen Verfahren ist für jede Ressourcenart eine zweiwertige Entscheidungsvorschrift abgespeichert. Diese Vorschriften sind den Ressourcenarten zugeordnet. Eine Entscheidungsvorschrift für die Ressourcenart Ra liefert für jedes Tupel, das aus

einer Anforderung und einer Bereitstellung einer Ressource der Art Ra besteht, entweder als Ergebnis die Aussage, daß die Anforderung kompatibel mit der Bereitstellung ist, oder die gegenteilige Aussage.

Wird ein technisches System ressourcenorientiert konfiguriert, so werden im Verlaufe der Konfigurierung — in der Regel wiederholt — die geforderten Ressourcen mit den bereitgestellten verglichen. Hierfür wird für jede Ressourcenart mindestens eine eigene Bilanz aufgestellt. Besteht das zu konfigurerende technische System aus mehreren räumlich getrennten Teilsystemen, so werden mehrere lokale Bilanzen für dieselbe Ressourcenart aufgestellt. Die Erfindung legt fest, daß dieser automatische Vergleich in den Bilanzen nicht nur hinsichtlich Art und Umfang, sondern auch hinsichtlich unterschiedlicher Qualitäten von Ressourcen durchgeführt wird. Daher besteht eine Bilanz für eine Ressourcenart, für die unterschiedliche Qualitäten definiert sind, aus mehreren Teilbilanzen, nämlich eine für jede Qualität einer bereitgestellten oder geforderten Ressource.

Die Bilanz ist ausgeglichen, wenn jede Teilbilanz es ist. Die Teilbilanz T1 für eine Ressourcenqualität Q1 ist ausgeglichen, wenn jede Ressourcen-Anforderung in T1 auf eine der folgenden beiden Bereitstellungen ausgeglichen ist:

- durch die Bereitstellung einer Ressource in T<sub>1</sub> selber
  - durch die Bereitstellung einer Ressource in einer anderen Teilbilanz T<sub>2</sub> derselben Ressourcenart.
- Dieser Ausgleich ist nur dann möglich,
- wenn T<sub>2</sub> für eine mit Q<sub>1</sub> kompatiblen Ressource steht
  - und wenn in der Teilbilanz T<sub>2</sub> ein Überschuß besteht, d. h. die Ressourcen-Bereitstellungen in T<sub>2</sub> nicht bereits durch die Ressourcen-Anforderungen in T<sub>2</sub> oder in einer dritten Teilbilanz verbraucht werden.

Im ersten Fall wird die Ressourcenanforderung also durch eine Ressource gleicher Qualität befriedigt im zweiten Fall durch eine kompatible Ressource.

Um die Qualitäten und Entscheidungsvorschriften zu definieren, legt man in der bevorzugten Ausführung der Erfindung fest, durch welche Attribute verschiedene Ressourcen derselben Art sich unterscheiden. Jedes Attribut steht für einen technischen Aspekt der Ressourcenart. Für jedes Attribut legt man fest, von welchem Datentyp die möglichen Werte des Attributs sind. Zusätzlich kann der Wertebereich eines Attributs definiert sein. Diese Definitionen und Festlegungen erfolgen für Ressourcenarten und unabhängig von den Komponententypen. Zwei Beispiele: Die Ressource "Strom bei 12-V-Gleichspannung" hat die Attribute "Ist-Spannung" und "Art der Absicherung gegen Überstrom?". Das erste Attribut ist stetig, das zweite diskret. "Steckplatz" hat keine Attribute.

Für ein Attribut kann man zusätzlich festlegen, welche Werte für das Attribut prinzipiell überhaupt möglich sind. Das Attribut "Art der Absicherung gegen Überstrom?" hat den Wertebereich {"Strombegrenzung", "Abschaltung", "fold-back"}.

Sei R eine Ressource von der Art Ra und sei festgelegt, daß Ra ein Attribut A besitzt. Im Katalog der verfügbaren Komponententypen ist für jeden Komponententyp K<sub>1</sub>, der die Ressource R bereitstellt eingetragen, welcher Bereich den Wert des Attributes A für die Komponenten vom Typ K<sub>1</sub> kennzeichnet. Dieser Be-

reich ist eine Teilmenge des Wertebereichs von A. Außerdem ist für jeden Komponententyp K<sub>2</sub>, der R fordert, notiert, in welchem Bereich der Wert von A liegen muß, damit eine Komponente des Typs K<sub>2</sub> ihre Sollfunktion erfüllt.

Für einen bestimmten Komponententyp wird man im Komponententypkatalog festlegen, daß bei einer Komponente dieses Typs der Wert eines stetigen Attribut einer Ressource in ein bestimmtes Intervall fällt bzw. fallen muß. Falls man einen genauen Wert festlegen will, definiert man ein Intervall mit einem Element. Für ein diskretes Attribut legt man eine Menge fest, die kein, ein oder mehrere Elemente enthält.

Zwei Beispiele: Für einen bestimmten Typ von Elektronik-Modulen, der Strom bei 12-V-Gleichspannung voraussetzt, ist notiert, daß die Ist-Spannung zwischen 10 V und 14 V liegen muß und die Stromversorgung bei Überstrom abgeschaltet werden muß. Ein bestimmter Typ von Spannungsversorgern liefert Strom bei 12-V-Gleichspannung mit einer garantierten Ist-Spannung zwischen 11 V und 13 V und mit einer Absicherung durch Strombegrenzung und durch Abschaltung.

Im folgenden werden vorteilhafte Ausgestaltungen der Erfindung beschrieben, um folgende Aufgabe zu lösen: Die unterschiedlichen Qualitäten einer Ressourcenart Ra seien durch n Attribute A<sub>1</sub>, ..., A<sub>n</sub> gekennzeichnet. Aufzustellen ist eine automatisch auswertbare Entscheidungsvorschrift  $\Phi$ , die zwei Ressourcen R<sub>1</sub> und R<sub>2</sub> der Ressourcenart Ra vergleicht. R<sub>1</sub> ist eine bereitgestellte und R<sub>2</sub> eine geforderte Ressource.

— Zwei Ressourcen R<sub>1</sub> und R<sub>2</sub> werden bezüglich jedes Attributes einzeln verglichen. R<sub>1</sub> ist nur dann mit R<sub>2</sub> kompatibel, wenn R<sub>1</sub> bezüglich jedes Attributes mit R<sub>2</sub> kompatibel ist. Formal: Man definiert n Funktionen  $\Phi_i$ . Jede einzelne Funktion  $\Phi_i$  ist für den Vergleich bezüglich des Attributes A<sub>i</sub> definiert ( $i = 1, \dots, n$ ) und liefert entweder "kompatibel" oder "nicht kompatibel" zu  $\Phi$  liefert "kompatibel", falls alle  $\Phi_i$  "kompatibel" liefern, ansonsten "nicht\_kompatibel".

— Der menschliche Entwickler, der die Informationen über die Ressourcenarten formuliert und einem Rechner verfügbar macht, spart Arbeit ein, wenn er für eine Ressourcenart mit den Attributen A<sub>1</sub>, ..., A<sub>n</sub> die Entscheidungsvorschriften  $\Phi_1, \dots, \Phi_n$  nicht vollständig neu definieren muß, sondern sie nur aus einem vorgegebenen Katalog von Vorschriften auswählt.

Die Entscheidungsvorschrift  $\Phi_i$  vergleicht dann, wenn das Attribut A<sub>i</sub> stetig ist, zwei Intervalle. Ist A<sub>i</sub> diskret, werden zwei Mengen verglichen. Der Entwickler definiert  $\Phi_i$ , indem er eine von zwei möglichen Entscheidungsvorschriften auswählt:

— Das Intervall bzw. die Menge I<sub>1</sub> (Werte der bereitgestellten Ressource) ist Teilmenge des Intervalls bzw. der Menge I<sub>2</sub> (Werte der geforderten Ressource). Diese Entscheidungsvorschrift ist für das Attribut Ist-Spannung anzuwenden.

— Das Intervall bzw. die Menge I<sub>1</sub> (Werte der bereitgestellten Ressource) ist Obermenge des Intervalls bzw. der Menge I<sub>2</sub> (Werte der geforderten Ressource).

Der Bereitsteller der Ressource Strom bei 12-V-Gleichspannung muß mindestens alle diejenigen Absicherungen gegen Überstrom bieten, die

der Verbraucher erfordert.

Eine weitere vorteilhafte Ausgestaltung der Erfindung spart Laufzeit bei der Ausführung einer Konfigurierung ein, indem identische Prüfungen auf Kompatibilität nicht mehrmals durchgeführt werden, sondern nur einmal erfolgen und ihr Ergebnis abgespeichert werden. Dies wird mit Hilfe einer Datenstruktur realisiert, die die Form von gerichteten Graphen hat. Pro Ressourcenart wird ein gerichteter Graph angelegt. Jeder Knoten des gerichteten Graphen steht für eine Qualität einer Ressource dieser Art. Der Knoten für die Ressource R<sub>1</sub> ist Nachfolger des Knotens für die Ressource R<sub>2</sub>, falls R<sub>2</sub> kompatibel mit R<sub>1</sub> ist.

15 Dieser gerichtete Graph wird vorteilhafterweise bereits dann erzeugt, wenn der Katalog verfügbarer Komponententypen aufgestellt ist. Er repräsentiert jede Qualitäten einer Ressource, die in diesem Katalog als von einem Komponententyp bereitgestellt oder gefordert ist. Um den Graphen aufzustellen, wird der Katalog vollständig nach Ressourcenqualitäten durchsucht. Jeder nur mögliche Vergleich zwischen zwei Ressourcen derselben Art mit verschiedenen Qualitäten wird durchgeführt und jede Kompatibilität als Kante in dem Graphen eingetragen.

20 Der gerichtete Graph braucht nur dann erweitert zu werden, wenn in einer Spezifikation eine Ressourcenqualität gefordert wird, die nicht im Komponentenkatalog genannt ist oder wenn der Komponententypkatalog um einen Komponententyp mit neuen technischen Qualitäten erweitert wird. Zweckmäßigerverweise ersetzt der "neue" Graph, der aufgrund einer neuen Spezifikation erweitert wurde, den alten Graphen.

25 Falls zur Laufzeit zu prüfen ist, ob eine bereitgestellte Ressource R<sub>1</sub> mit einer geforderten Ressource R<sub>2</sub> kompatibel ist, braucht die Informationsverarbeitungsanlage nur noch die beiden Knoten A und B im Graphen zu suchen, die für R<sub>1</sub> bzw. R<sub>2</sub> stehen. Falls eine Kante von A nach B existiert, falls also A ein Vorgänger von B und B ein Nachfolger von A ist, so ist R<sub>1</sub> mit R<sub>2</sub> kompatibel, ansonsten nicht. Da in der Regel mit demselben Komponententypkatalog mehrere Konfigurierungsaufgaben auszuführen sind, ist der nur einmal anfallende Rechenzeitaufwand für das Aufstellen des Graphen geringer als der zur Laufzeit eingesparte Rechenzeitaufwand für die Kompatibilitätsprüfungen.

30 Eine weitere vorteilhafte Ausgestaltung liegt darin, daß bereits nach Aufstellung des Komponententyp-Kataloges unabhängig von einer bestimmten Spezifikation geprüft wird, ob die im Katalog beschriebenen Komponenten zueinander passen. Für jede Ressource, die ein im Katalog berücksichtigter Komponententyp K<sub>1</sub> in einer bestimmten Qualität fordert, wird nach einem Komponententyp K<sub>2</sub> gesucht, der die Ressource in der gleichen Qualität oder eine kompatible Ressource bereitstellt. Liefert diese Suche kein Ergebnis, so steht fest, daß bei einer automatischen Konfigurierung niemals eine Komponente des Typs K<sub>1</sub> ausgewählt werden darf. Ansonsten wären die Ressourcenanforderungen dieser Komponente nicht zu erfüllen. Bei einer automatischen Konfigurierung müßte die Entscheidung für eine Komponente des Typs K<sub>1</sub> zurückgezogen werden, ansonsten würde die Konfigurierung scheitern. Zweckmäßigerverweise ergänzt ein menschlicher Entwickler den Katalog verfügbarer Komponententypen oder ändert ihn so ab, daß dieses Ergebnis nicht mehr auftreten kann.

35 Eine andere Art von Vorab-Prüfung erkennt nach Aufstellen der formalen Spezifikation, aber noch vor

der eigentlichen Konfigurierung, daß die formale Spezifikation nicht erfüllbar ist. Von Vorteil ist es, dieses negative Ergebnis so früh wie möglich zu ermitteln und geeignete Abhilfen zu treffen. Für jede Ressource, die in einer bestimmten Qualität vom technischen System gefordert wird, wird nach einem geeigneten Komponententyp im Katalog gesucht. Läßt sich keiner finden, so steht bereits vor der eigentlichen Konfigurierung fest, daß mit den verfügbaren Komponenten kein technisches System konfiguriert werden kann, das der Spezifikation genügt.

Die beiden letztgenannten Ausgestaltungen lassen sich kombinieren. Der oben beschriebene Graph wird automatisch erzeugt. In jedem seiner Knoten wird notiert, ob die repräsentierte Ressourcenqualität eine bereitgestellte oder eine geforderte oder beides ist. Dann werden die Wurzeln des Graphen durchsucht, also alle Knoten, die keine Vorgänger haben. Wird hierbei eine Wurzel gefunden, die für eine geforderte Ressourcenqualität steht, so steht fest, daß eine Anforderung nach dieser Ressourcenqualität nicht erfüllt werden kann.

Zweckmäßigerweise werden in den drei Speichern Informationen, die für mehrere Komponententypen  $K_1, \dots, K_n$  gelten, nur einmal abgespeichert. Bei den  $n$  Komponententypen werden nicht mehr diese Informationen gespeichert, sondern nur noch eine Vorschrift, wie diese Informationen automatisch beschafft werden können. Dieses Vorgehen spart Speicherplatz, außerdem wird Zeit bei der Aufstellung der Informationen gespart.

Ein üblicher und anschaulicher Weg, den der Stand der Technik kennt, ist das Aufstellen einer Taxonomie (= Verwandtschaftshierarchie) unter Komponententypen. Beim Aufstellen des Katalogs verfügbarer Komponententypen führt man abstrakte Oberklassen von Komponententypen an, beispielsweise "Steckkarte". Die verfügbaren Komponententypen sind dann Unterklassen dieser abstrakten Oberklassen. Für mehrere abstrakte Oberklassen kann man wiederum eine gemeinsame (Ober-)Oberklasse einführen, die Taxonomie kann beliebig viele Ebenen haben.

Informationen, die allen Unterklassen einer Oberklasse gemeinsam sind, werden der Oberklasse zugeordnet. Beispielsweise wird für den Komponententyp "Steckkarte" notiert, daß jede Steckkarte die Ressource "Steckplatz" benötigt. Diese Informationen gelten damit für alle Unterklassen. Aus der Technik des objekt-orientierten Programmierens sind Verfahren bekannt, wie eine Unterklassse von einer Oberklasse Informationen "erbt". Verfeinerungen lehren Wege des mehrfachen Vererbens ("multiple inheritance"), bei der eine Klasse mehrere Oberklassen hat, von denen sie Informationen erbtt, sowie Verfahren, um die gesamten Informationen über eine Klasse automatisch aus Informationen, die für die Klasse spezifisch sind und daher bei der Klasse selbst notiert sind, und aus Informationen, die bei einer Oberklasse notiert sind, zusammenzusetzen. In einer vorteilhaften Ausführung der Erfahrung werden diese Verfahren angewendet, um beim Aufstellen des Katalogs verfügbarer Komponententypen Arbeitszeit und Speicherplatz zu sparen.

Nicht nur die Komponententypen lassen sich in einer Taxonomie anordnen, sondern auch für die Ressourcenarten stellt man vorteilhafterweise eine eigene Taxonomie auf. Vererbt werden kann beispielsweise die Existenz, der Datentyp und der Wertebereich von Attributen.

Den Katalog verfügbarer Komponententypen stellt ein menschlicher Experte einmal auf. Er definiert hierbei

die Ressourcenarten und legt fest, welche Ressourcen und Ressourcenqualitäten die Komponententypen bereitstellen bzw. fordern. Die Anforderungsspezifikation an das zu konfigurierende technische System formuliert ein Projektierer. Beide Personen bedienen sich zweckmäßigerweise einer leistungsfähigen Benutzeroberfläche, wie sie Stand der Technik ist. In einer vorteilhaften Ausgestaltung legt der Experte einmal fest, welche Werte ein Attribut einer Ressourcenart überhaupt annehmen können. Dieser Wertebereich ist für ein diskretes Attribut typischerweise eine Menge von Symbolen, beispielsweise {"Strombegrenzung", "Abschaltung", "fold-back"} für "Art der Absicherung gegen Überstrom?". Für ein stetiges Attribut wird der Experte meist ein Intervall definieren. Diese Festlegung vereinfacht die weitere Arbeit: Wenn festzulegen ist, in welcher Qualität ein Komponententyp eine durch Attribute gekennzeichnete Ressource bereitstellt oder erfordert, so bietet die Benutzeroberfläche dem Experten nur die Werte des zuvor festgelegten Wertebereiches eines diskreten Attributes zur Auswahl an. Bei einem stetigen Intervall muß der Zahlenbereich in das Intervall, das den Wertebereich kennzeichnet, fallen, was automatisch geprüft wird – ansonsten hat der Experte einen Fehler begangen. Die Festlegung der Wertebereiche unterstützt den Projektierer in analoger Weise dabei, die Anforderungsspezifikation für ein zu konfigurierendes technisches System zu beschreiben.

#### Patentansprüche

1. Verfahren zur automatischen Konfigurierung von technischen Systemen aus Komponenten unter Verwendung einer Informationsverarbeitungseinheit

mit Lesezugriff auf einen ersten Speicher (S1),

- in dem ein Katalog verfügbarer Komponententypen abgespeichert wird
- und dabei für jeden verfügbaren Komponententyp im ersten Speicher (S1) mindestens eine der folgenden Informationen abgelegt wird:

– Art und Umfang der von einer Komponente des Typs bereitgestellten Ressourcen

Art und Umfang der von einer Komponente des Typs vorausgesetzten, verbrauchten oder belegten Ressourcen,

mit Lesezugriff auf einen zweiten Speicher (S2),

– in dem Informationen über Ressourcenarten abgespeichert werden, die jeweils allen Ressourcen derselben Art gemeinsam sind, und mit Lesezugriff auf einen dritten Speicher (S3),

- in dem die Anforderungsspezifikation für das zu konfigurierende technische System abgespeichert wird

– und dabei in diesem Speicher Art und Umfang der für das technische System geforderten Ressourcen abgelegt wird,

wobei die automatische Auswahl von Komponenten für das technische System durch die Informationsverarbeitungseinheit so vorgenommen wird, daß, ausgehend von den für das technische System geforderten Ressourcen, solange noch Ressourcenbedarf besteht und Komponententypen im Katalog existieren, welche entsprechende Ressourcen bereitstellen,

wiederholt jeweils eine oder mehrere Komponen-

ten des bestgeeigneten Komponententyps ausgewählt werden und deren bereitgestellte Ressourcen bzw. deren vorausgesetzte oder verbrauchte oder belegte Ressourcen im verbleibenden Ressourcenbedarf berücksichtigt werden,  
dadurch gekennzeichnet,  
daß im zweiten Speicher (S2)

- für mindestens eine Ressourcenart
  - die Information abgespeichert wird, welche unterschiedlichen Qualitäten die Ressourcen dieser Art kennzeichnen
  - und eine Entscheidungsvorschrift notiert wird, die unter Bezugnahme auf mindestens eine dieser Qualitäten die Bedingung festlegt, unter denen eine bereitgestellte Ressource dieser Art mit einer verbrauchten/belegten/vorausgesetzten Ressource dieser Art kompatibel ist,

daß im ersten Speicher (S1) 20

- für mindestens einen Komponententyp die Information abgelegt wird,
  - von welcher Qualität eine von einer Komponente des Typs bereitgestellte Ressource ist oder
  - von welcher Qualität eine von einer Komponente des Typs verbrauchte/belegte/vorausgesetzte Ressource sein muß, damit die Komponente die von ihr geforderte Funktion erfüllt,

und daß bei der automatischen Auswahl von Komponenten für das technische System mindestens einmal dann, wenn der jeweils verbleibende Ressourcenbedarf bestimmt wird,

- für jede Ressource, die von den im bisherigen Verlauf der Konfigurierung ausgewählten Komponenten vorausgesetzt oder verbraucht oder belegt wird und die von einer Ressourcenart ist, die gemäß den Informationen, die im zweiten Speicher (S2) abgespeichert werden, durch unterschiedliche Qualitäten gekennzeichnet ist,

geprüft wird, ob eine Ressource von gleicher Qualität wie die geforderte Ressource oder eine Ressource, die mit der geforderten Ressource kompatibel ist, von den im bisherigen Verlauf der Konfigurierung ausgewählten Komponenten bereitgestellt wird, wobei immer dann, wenn eine bereitgestellte Ressource auf Kompatibilität mit der geforderten Ressource geprüft wird, automatisch Entscheidungsvorschriften angewendet werden, die im zweiten Speicher (S2) abgespeichert sind,

und dann, wenn noch Ressourcenbedarf besteht, 55  
— mindestens eine Komponente ausgewählt wird, die die geforderte Ressource in der gleichen Qualität bereitstellt oder die eine mit der geforderten Ressource kompatible Ressource bereitstellt.

2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, daß im dritten Speicher (S3)

- für mindestens eine Ressource, die gemäß der Anforderungsspezifikation vom zu konfigurierenden technischen System gefordert ist, notiert wird, von welcher Qualität diese Ressource sein muß.

3. Verfahren nach Anspruch 1 oder Anspruch 2,

dadurch gekennzeichnet, daß im zweiten Speicher (S2)

- für mindestens eine Ressourcenart
  - die Information abgespeichert wird, welche Attribute die Qualitäten der Ressourcen dieser Art kennzeichnen,
  - die Information abgespeichert wird, von welchen Datentypen die Attribute sind,

— und die Entscheidungsvorschrift, die die Bedingung festlegt, unter denen eine bereitgestellte Ressource dieser Art mit einer verbrauchten/belegten/vorausgesetzten Ressource dieser Art kompatibel ist, auf diese Attribute Bezug nimmt,

daß im ersten Speicher (S1)

- für mindestens einen Komponententyp
  - die Information abgespeichert wird, welche Werte die Attribute der von einer Komponente dieses Typs bereitgestellten Ressourcen annehmen oder
  - die Information abgespeichert wird, welche Werte die Attribute der von einer Komponente dieses Typs vorausgesetzten oder verbrauchten oder belegten Ressourcen annehmen müssen.

4. Verfahren nach Anspruch 3, dadurch gekennzeichnet, daß im zweiten Speicher (S2)

- für mindestens eine Ressourcenart die zugeordnete Entscheidungsvorschrift daraus besteht,  
daß eine bereitgestellte Ressource einer bestimmten Qualität dann kompatibel mit einer geforderten Ressource einer anderen Qualität dieser Ressourcenart ist,  
wenn die Menge der Werte, die ein Attribut der Ressourcenart bei der Qualität der bereitgestellten Ressource annimmt, eine Teilmenge der Menge der Werte ist, die ein Attribut der Ressourcenart bei der Qualität der geforderten Ressource nur annehmen darf.

5. Verfahren nach Anspruch 3 oder Anspruch 4, dadurch gekennzeichnet, daß im zweiten Speicher (S2)

- für mindestens eine Ressourcenart die zugeordnete Entscheidungsvorschrift daraus besteht,  
daß eine bereitgestellte Ressource einer bestimmten Qualität dann kompatibel mit einer geforderten Ressource einer anderen Qualität dieser Ressourcenart ist,  
wenn die Menge der Werte, die ein Attribut der Ressourcenart bei der Qualität der bereitgestellten Ressource annimmt, eine Obermenge der Menge der Werte ist, die ein Attribut der Ressourcenart bei der Qualität der geforderten Ressource nur annehmen darf.

6. Verfahren nach Anspruch 3 oder Anspruch 4, dadurch gekennzeichnet, daß im zweiten Speicher (S2)

- für mindestens eine Ressourcenart die Information abgespeichert wird, welche Werte die Attribute, die die Ressourcen dieser Art kennzeichnen, annehmen.

7. Verfahren nach Anspruch 6, dadurch gekennzeichnet,

- daß weitere Informationen über die Ressourcen, die ein verfügbarer Komponententyp bereitstellt oder fordert,  
durch Interaktion mittels einer Bedienoberfläche gewonnen und im ersten Speicher (S1) abgelegt werden  
und die Bedienoberfläche hierbei die Werte eines Attributes einer Ressource zur Auswahl anbietet, wobei  
die Bedienoberfläche nur diejenigen Werte anbietet, die im zweiten Speicher (S2) als Wertebereich des Attributes abgelegt sind. 10
8. Verfahren nach Anspruch 6 oder Anspruch 7, dadurch gekennzeichnet,  
daß weitere Informationen über eine Ressource, die vom zu konfigurierenden technischen System gefordert wird, durch Interaktion mittels einer Bedienoberfläche gewonnen und im dritten Speicher (S3) abgelegt werden 20  
und die Bedienoberfläche hierbei die Werte eines Attributes der Ressource zur Auswahl anbietet, wobei  
die Bedienoberfläche nur diejenigen Werte anbietet, die im zweiten Speicher (S2) als Wertebereich 25 des Attributes abgelegt sind.
9. Verfahren nach einem der Ansprüche 1 bis 8, dadurch gekennzeichnet,  
daß Informationen über die Entscheidungsvorschriften in einem ersten Teilspeicher (S2a) des zweiten Speichers (S2) abgelegt werden  
und daß die übrigen Informationen über die Ressourcenarten in einem zweiten Teilspeicher (S2b) des zweiten Speichers (S2) abgelegt werden  
und daß im zweiten Teilspeicher (S2b) für jede Ressourcenart die Information abgespeichert ist, 35  
— welche der Entscheidungsvorschriften, über die im ersten Teilspeicher (S2a) Informationen abgelegt sind, für eine Prüfung auf Kompatibilität von Ressourcen dieser Ressourcenart angewendet wird. 40
10. Verfahren nach einem der Ansprüche 1 bis 9, dadurch gekennzeichnet, daß für mindestens eine Ressourcenart (R)  
— im zweiten Speicher (S2) mehrere Teilentscheidungsvorschriften abgelegt werden 45  
— und im zweiten Speicher (S2) die Information abgelegt wird,  
daß die Entscheidungsvorschriфт für die Ressourcenart (R) daraus besteht,  
— daß eine bereitgestellte Ressource dann mit einer geforderten Ressource dieser Ressourcenart kompatibel ist, 50  
wenn die Anwendung jeder Teilentscheidungsvorschriфт das Ergebnis liefert, daß die bereitgestellte Ressource kompatibel mit der geforderten Ressource ist,  
— und daß eine bereitgestellte Ressource dann mit einer geforderten Ressource dieser Ressourcenart nicht kompatibel ist, 55  
wenn die Anwendung mindestens einer Teilentscheidungsvorschriфт das Ergebnis liefert, daß die bereitgestellte Ressource nicht kompatibel mit der geforderten Ressource ist. 60
11. Verfahren nach einem der Ansprüche 3 bis 8, dadurch gekennzeichnet, daß im zweiten Speicher (S2) 65

- für jedes Attribut mindestens einer Ressourcenart (R), für die mehrere Attribute definiert sind, mindestens eine Teilentscheidungsvorschriфт abgespeichert wird, die nur auf dieses Attribut Bezug nimmt, und die Entscheidungsvorschriфт für die Ressourcenart (R) daraus besteht,
  - daß eine bereitgestellte Ressource dann mit einer geforderten Ressource dieser Ressourcenart kompatibel ist, wenn die Anwendung jeder Teilentscheidungsvorschriфт das Ergebnis liefert, daß die bereitgestellte Ressource bezüglich des Attributes, auf das diese Teilentscheidungsvorschriфт Bezug nimmt, kompatibel mit der geforderten Ressource ist
  - und daß eine bereitgestellte Ressource dann mit einer geforderten Ressource dieser Ressourcenart nicht kompatibel ist, wenn die Anwendung mindestens einer Teilentscheidungsvorschriфт das Ergebnis liefert, daß die bereitgestellte Ressource bezüglich des Attributes, auf das diese Teilentscheidungsvorschriфт Bezug nimmt, nicht kompatibel mit der geforderten Ressource ist.
12. Verfahren nach einem der Ansprüche 1 bis 11, dadurch gekennzeichnet, daß für mindestens eine Ressource,  
— deren Bereitstellung eine Komponente eines ersten Komponententyps (K1), über den Informationen im ersten Speicher (S1) abgespeichert sind, in einer bestimmten Qualität fordert,  
— geprüft wird,  
ob für den ersten Komponententyp (K1) oder einen zweiten Komponententyp im ersten Speicher (S1) die Information abgespeichert ist,  
daß eine Komponente dieses Typs entweder diese Ressource in der geforderten Qualität oder eine kompatible Ressource bereitstellt, und dann, wenn kein Komponententyp gefunden wird, über den diese Information abgespeichert ist,  
— der erste Komponententyp (K1) so gekennzeichnet wird, daß bei einer automatischen Konfiguration eines technischen Systems eine Komponente dieses Typs nicht ausgewählt werden kann  
— und bei Bedarf diese Information über den ersten Komponententyp (K1) ausgegeben wird.
13. Verfahren nach einem der Ansprüche 1 bis 12, dadurch gekennzeichnet, daß für mindestens eine Ressource,  
— die in der Anforderungsspezifikation, die im dritten Speicher (S3) abgelegt ist, in einer bestimmten Qualität gefordert wird,  
— geprüft wird,  
ob für einen Komponententyp im ersten Speicher (S1) die Information abgespeichert ist, daß eine Komponente dieses Typs entweder diese Ressource in der geforderten Qualität oder eine kompatible Ressource bereitstellt, und dann, wenn kein Komponententyp gefunden wird, über den diese Information abgespeichert ist,  
— die Information ausgegeben wird,

daß die Anforderungsspezifikation, die im dritten Speicher (S3) abgelegt ist,  
mit Komponenten der Komponententypen,  
über die im ersten Speicher (S1) Informationen  
abgespeichert sind,  
nicht erfüllt werden kann,  
— und die Ressource(n), die nicht bereitgestellt werden können,  
bei Bedarf gekennzeichnet werden.

14. Verfahren nach einem der Ansprüche 1 bis 13, 10  
dadurch gekennzeichnet, daß für mindestens eine Ressourcenart

- für mindestens zwei Ressourcen (R1, R2)  
dieser Art von unterschiedlichen Qualitäten,
  - von denen die erste Ressource (R1) von 15  
einem Komponententyp bereitgestellt wird, über den im ersten Speicher (S1) Informationen abgelegt sind,
  - und die zweite Ressource (R2) von einem Komponententyp gefordert wird, 20  
über den im ersten Speicher (S1) Informationen abgelegt sind, oder von der Anforderungsspezifikation gefordert wird, die im dritten Speicher (S3) abgespeichert ist, im zweiten Speicher (S2) die Information 25  
abgespeichert wird, daß die erste Ressource (R1) mit der zweiten Ressource (R2) kompatibel ist,

und daß diese Information bei der automatischen Auswahl von Komponenten für das technische System mindestens einmal dann, wenn die zweite Ressource (R2) gefordert wird, berücksichtigt wird. 30

15. Verfahren nach Anspruch 14, dadurch gekennzeichnet, daß in einer ersten Phase

- für jede Ressourcenart, über die im zweiten 35  
Speicher (S2) Informationen abgespeichert werden,  
für jedes Paar, das aus zwei Ressourcen dieser Art von unterschiedlichen Qualitäten besteht,
  - von denen die erste Ressource (R1) von 40  
einem Komponententyp bereitgestellt wird, über den im ersten Speicher (S1) Informationen abgelegt sind,
  - und die zweite Ressource (R2) von einem Komponententyp gefordert wird, 45  
über den im ersten Speicher (S1) Informationen abgelegt sind,  
im zweiten Speicher (S2) die Information abgespeichert ist, daß die erste Ressource (R1) mit der zweiten Ressource (R2) kompatibel ist, 50

und in einer zweiten Phase

- dann, wenn die automatische Konfigurierung eines technischen Systems durchgeführt werden soll, 55  
die in der ersten Phase abgespeicherten Informationen bei der automatischen Auswahl von Komponenten für das technische System berücksichtigt werden.

16. Verfahren nach einem der Ansprüche 1 bis 15, 60  
dadurch gekennzeichnet, daß im ersten Speicher (S1)

- Informationen, die für jeden von mehreren Komponententypen gelten, nur einmal abgespeichert werden,
- und für jeden dieser Komponententypen eine Vorschrift notiert wird, 65  
wie diese Informationen bei der automati-

schen Konfigurierung eines technischen Systems beschafft werden können.

17. Verfahren nach einem der Ansprüche 1 bis 16, dadurch gekennzeichnet, daß im zweiten Speicher (S2)

- Informationen, die für jede von mehreren Ressourcenarten gelten, nur einmal abgespeichert werden
- und für jede dieser Ressourcenarten eine Vorschrift notiert wird, wie diese Informationen bei der automatischen Konfigurierung eines technischen Systems beschafft werden können.

18. Verfahren nach Anspruch 16,

daß die Komponententypen des Katalogs in einer Taxonomie der Komponententypen hierarchisch unter Oberklassen geordnet werden, indem zu jedem Komponententyp im ersten Speicher (S1) ein Verweis auf die Oberklasse und zu jeder Oberklasse ein Verweis auf deren Oberklasse abgespeichert wird, daß Informationen, die für alle Komponententypen, die einen Verweis auf dieselbe Oberklasse tragen, gelten,

bei dieser Oberklasse im ersten Speicher (S1) notiert werden, und daß Informationen über einen Komponententyp, die nicht bei dem Komponententyp selber notiert sind,

den gespeicherten Daten einer Oberklasse des Komponententyps entnommen werden.

19. Verfahren nach Anspruch 17,

daß die Ressourcenarten in einer Taxonomie der Ressourcenarten hierarchisch unter Oberklassen geordnet werden,

indem zu jeder Ressourcenart im zweiten Speicher (S2) ein Verweis auf die Oberklasse und zu jeder Oberklasse ein Verweis auf deren Oberklasse abgespeichert wird,

daß Informationen, die für alle Ressourcenarten, die einen Verweis auf dieselbe Oberklasse tragen, gelten,

bei dieser Oberklasse im zweiten Speicher (S2) notiert werden und daß Informationen über eine Ressourcenart, die nicht bei der Ressourcenart selber notiert sind,

den gespeicherten Daten einer Oberklasse der Ressourcenart entnommen werden.

**- Leerseite -**

**THIS PAGE BLANK (USPTO)**